

GRAPH BASED CLASSIFIER TO DETECT MALICIOUS URL

JAYAKANTHAN. N¹ & A. V. RAMANI²

¹Department of Computer Applications, Kumaraguru College of
Technology, Coimbatore, Tamil Nadu, India

²Department of Computer Science, Sri Rama Krishna Mission Vidyalaya
College of Arts and Science, Coimbatore, Tamil Nadu, India

ABSTRACT

Malicious attack is a major issue in cyberspace. The criminal obtains vital information like username, password, and Credit/Debit card numbers, from the victims through deception. Various detection solutions are proposed in recent years. These techniques include blacklist, heuristics, machine learning, similarity and pattern matching methods. But, most of them are heavy weight methodologies in terms of time complexity and requires dedicated server for their execution. A Graph based Classifier to Detect Malicious URL (GCDMU), is proposed in this paper, which is a feature based classifier. It is a light weight, reliable approach and also effective, in detecting malicious URL.

KEYWORDS: Malicious URL, Graph Based Classifier & Detection

Received: Aug 20, 2017; **Accepted:** Sep 10, 2017; **Published:** Sep 19, 2017; **Paper Id.:** IJMPERDOCT201724

INTRODUCTION

The internet is a vital component of human life. Large numbers of applications are added in webs, which may be accessed through the URL. The criminals use the fake URL, to redirect the user to malicious websites. The user who visits those websites becomes a victim of the cyber attack. It performs various activities like credential theft, extortion of money, and spreading viruses and worms [3]. The phishing attacks are with customized emails, with personal information to draw the individuals to specific Trojan-horse websites [4]. The increasing number of malicious attacks and large economic losses warns that, malicious URLs are not only a threat to individuals, but also for corporations and governments, in this cyber era. Malicious URL links are used to carry out phishing attacks, which steal user credentials and other significant information. The first phishing attack was detected in 1996, as a cyber attack. 123,555 phishing sites that are detected between October 2015- March 2016. According to APWG [3] report, the payment and financial services are the most affected by this type of attack.

This paper analyses various malicious URL detection methods, to identify their fault. The drawback of the existing approaches includes, the inability to detect the emerging attacks; limited features used for analysis and rely on blacklisted profile for classification. This paper proposes a graph based classification technique, to detect the malicious URL. This approach improves the accuracy and reduces the performance overhead. The core idea behind this approach is, the correct sets of features are used to detect the nature of the URL, and precise classification through graph based detection. The graph is formed using malicious features of the URL.

The highlights of this paper are listed below.

- Light weight method with significant features is used to reduce the computation complexity.
- The performance overhead is very less.
- No special resource like dedicated server is required for the proposed system.
- The graph based detection method improves the accuracy of the classification.

The paper is organized as follows. Section 2, describes the existing techniques and their drawbacks. Section 3 provides an overview of the features. The detailed description of the proposed system is given in section 4. Section 5 gives the analysis of the result. Section 6 concludes the paper.

RELATED WORK

The classification techniques are classified into three categories - URL based, Heuristics based and Machine learning based. Cuiwen Xiong et al [8], proposed an approach of malicious URL detection, using trigrams-based common pattern of URL, implanted with random domain recognition, named MIRD. The domain name, path and file names are analyzed. A common pattern, based on inverted index is used, to compare with the detected URL. Inability to detect the emerging attack, is the drawback of this approach. Astorino et al [5], developed a binary classification method, which analyzes the syntax and domain properties of the URL, using Support Vector Machine. Boyr-Moore, pattern matching method [11] is used to compare URL source code with virus characters in the database, to classify the URL to be safe or not. Anjali B. Sayamber and Arati M. Dixit [2], developed an approach to detect the malicious URL using Navie Bays algorithm. But, our proposed approach using graph based detection, results in better accuracy than this approach. Jian Cao et al [16], proposed a malicious URL detection method, through forwarding based features. Malicious websites using another URL feature are not detected in this algorithm. Alexandre Gerber et al [1], developed a method which analyzes the traffic pattern associated with URL, to detect malicious websites. Jayakanthan and A. V. Ramani [13] proposed feature based framework, to detect the malicious URL. It analyses the domain, sub domain, path and malicious characters to detect the malicious URL. ID3 decision tree algorithm is used for this purpose.

Chong et al, developed an approach which analyses lexical, source code and URL features using SVM, with polynomial kernel to detect malicious URL [7]. Dom Kim et al [10], developed a method to identify malicious URLs of biomedical information system. This work dynamically estimates the risk index of the affected URL, by analyzing node characteristics, malicious code, IP address and country code. Tung-Ming Koo [21] et al, developed a model which analyses the signature of the URL for classification. URL pattern matching method is used for this purpose.

Charmi Patel [6], proposed an approach that analyzes the lexical and network based features, to classify the nature of the URL using malicious URLs matching functions. A static code analysis technique was used; to detect the malicious URL, proposed by Prabhu Seshagiria et al [17]. It analyses the domain, iframe and image, to classify the URL. Hiba Zuhair et al [12], proposed a phish prediction method, which analyses 48 hybrid features, using support vector machine to detect the malicious URL. Time complexity and limitation, in detecting emerging attacks are the major drawbacks of this approach. Suyeon Yoo and Sehun Kim [20], proposed two phase classification to detect malicious web pages.

Urvashi Prajapat et al [22], proposed a repeated pruning approach, which analyses the URL and webpage features. RIPPER algorithm is used for the classification. Ralph Edem Agbefe [18] proposed domain information based blacklisting method, that compares the domain properties with a blacklist. This approach is capable to detect traditional attack, but not

able to detect emerging attacks. This paper proposes the model to identify the phishing site. N. Jayakanthan and A.V. Ramani [14] proposed a two phase classification model, to detect the malicious URLs. An Enhanced Probing Classification, to Detect Malicious URL (EPCMU) is used with Navie Bayes classifier, to detect malicious URLs. Jaydeep Solanki et al [15] proposed a decision tree based feature analysis method, to detect phishing attacks. The decision tree algorithm is used for classification. Roshani. K [19] et al developed a model, for the detection of malicious URL in twitter, using posterior probabilities of the tweet. A binary classification method is proposed by Astorino. A [6] et al and analyses the URL Syntax and domain properties, using spherical separation-based algorithm. Da Huang [9] et al used to dynamically extract lexical patterns from URL, for analysis and classification.

FEATURES

The proposed algorithm is a feature based classification method, to detect the malicious URL, using graph classifier. The features are given in the table 1. The proposed approach is a lightweight approach; the significant features are selected to distinguish the genuine and malicious URLs. The features are given as input to the classification algorithm, to identify the given URL as genuine or malicious.

Table 1: List of Features

SL. NO	Name of the Feature
1	Number of slashes
2	Number of Dots in the URL
3	Malicious Special Characters
4	Path tokens
5	@ symbol

Number of Slashes

The URL with five or more slashes is considered as malicious. The slash count is a significant character, to detect the malicious URL.

Number of Dots in URL

The malicious URL uses a number of dots in the domain and file path. If more than five dots are found in the URL, then it is considered as malicious.

Malicious Special Characters

The set (S) of special characters are not a part of the genuine URL, but, used for malicious attacks. The list of characters is given below

$$S = \{!,',\#, \$, '\wedge', '*', '(', ')', '+', '{', '}'\}$$

Path Tokens

The malicious tokens are used in the URL path, to bypass the security of the webserver. The occurrence of such token/s in the URL path is considered as malicious. The malicious token is “config”, “secure”, “.. /”, “password”, “.. %u2216”.

@ Symbol

The malicious URLs are embedded with the genuine URL, using @ symbol. It redirects the user, to the malicious

website. So the occurrence of such symbol in URL is considered as malicious.

METHODOLOGY

The proposed approach analyses the URL and classify as genuine or malicious. The Graph based Classifier to Detect Malicious URL (GCDMU), is used to detect the malicious URL. The URL entered in the browser is the input. Feature extractor collects the features; the features are grouped into a feature vector. GCDMU classifier analyses the features. If a malicious feature occurs, it is added as the node of the graph. The graph nodes represent the set of malicious features in the URL. If no malicious feature is present, then the graph is NULL. It represents the URL as genuine. The graphical approach is stable, than the tree based classifiers. The variations in the input instance require minor modification, in node of the graph that represents the particular feature. In the tree, it requires major changes and sometimes the entire tree needs to be restructured.

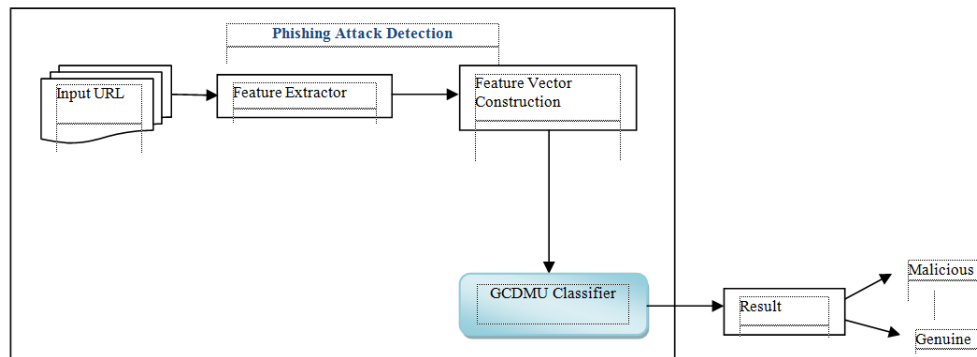


Figure 1: System Architecture

The extracted features are used as input to the classifier. The purpose of proposed work, is to distinguish the malicious and genuine URL.

Algorithm

Graphs are simple and easy to understand. It is adaptable to different kinds of problem. The graph is used to solve various problems in real time. We proposed a graph based algorithm, to detect the malicious URL. The F represents a feature vector, which contains the set of malicious features $f_1, \dots, f_n$. G is the graph. If the malicious feature is present, it is added as the node of the graph. After analyzing the URL, the nodes of the graph G, represent the set of malicious feature in the URL. If the graph is NULL, then the URL is declared as genuine. The proposed algorithm is tested against various data sources.

Proposed Approach

Input: The URL

Output: Classification of the URL is genuine or malicious.

- Analyze the Input URL.
- Count the number of dots and slashes in the URL

- If they are ≥ 5 then set the status of the URL is malicious.
- Compare the URL characters and malicious special character
 - If match found then the URL is malicious.
- Analyze the URL of the list of path tokens
 - Occurrence of the path tokens leads to classify the URL is malicious.
- Search for the any @ symbol present in the URL
 - If @ Symbol occurs consider the URL is malicious.
- Any malicious feature present then report the URL is malicious & list the Malicious features.
- No malicious feature is present, then declares the URL is genuine.

The Graph based Classification Algorithm, to Detect Malicious URL is developed. It constructs the graph, if the malicious feature is present in the URL. The Malicious Feature Collection (MFC) algorithm is developed, to collect the list of malicious features from the graph. The GCDMU algorithm constructs the graph for the malicious URL, and calls MFC to collect the malicious features from the graph.

Proposed Algorithm:

Graph Based Classifier to Detect Malicious URL (GCDMU)

// **Input:** The URL is to be tested.

//**Output:** The URL is malicious or Genuine.

$F = \{f_1 \dots f_n\}$ is the set of features

D is the decision result may be genuine or malicious.

// M_T is the set of malicious features

//G is the graph

Assign $G \leftarrow \text{Null}$

for $i \leftarrow 1$ to $|F|$ **do**

if a feature f^* is malicious and $\notin G$

then

$G \leftarrow G \cup \{f^*\}$

end for

If $G \neq \text{Null}$

Display the URL is malicious

Select a vertex any vertex v of the graph G arbitrarily

```

 $M_T \leftarrow \emptyset$  // Intialize the set  $M_T$  as Null

MFC( $v, M_T$ ) // Call malicious feature collection algorithm

Display the set  $M_T$  of Malicious Features

Else

    Display the URL is genuine

    The Malicious Feature Collection Algorithm, used by GCDMU to collect the malicious feature of the graph is
    given below.

```

Algorithm MFC (V, M_i)

//visits recursively all the unvisited nodes from the input node v by a path , number them in //order they are encountered via global variable count and then collect the malicious features in a //set M_T .

Input: The vertex V , The set M_T // Graph vertex v , malicious feature set M_T

Output: M_T , the set of nodes contain malicious features

count $\leftarrow$ count+1; mark v with count

$M_T \leftarrow M_T \cup v$

for each vertex u in V adjacent to a v **do**

if u is marked with 0 then

MFC(u)

end for

ANALYSIS OF THE URL

The list of URLs is given below.

- <http://recipa^1.com@74.125.131.105>
- <http://www.botts.com/wp-admin/espacecaf.fr/0FZEFZEF0ZEFZEF0ZEF0EZFEZFZE508F5ZE8F04EZFO48ZEF48EZ0F48ZEF/3dsecureclient.service.caf.fr/id/eec5bc2a4a47e2941bbac5a62f3e979e/ password step1.htm>
- <http://www.figuin.com>
- <http://www.sarvota.org.ar/ss-admin/ secur /espace-caf.fr.2dsecureclient@.service.caf.fr>
- <http://www.avokka.org.ar/Client/espacecaf.fr/0FZEFZEF0ZEFZEF0ZEF0EZFEZFZE508F5ZE8F04EZFO48ZEF48EZ0F48ZEF/3dsecureclient.service.caf.fr/id/2ee96247eb996860335d0d28bcf6cc09/step3config.htm>
- [http://www.brueacs.com/wp-admin/load- c=1&load\[\]=swfobject.jquery.utils&ver=3.5](http://www.brueacs.com/wp-admin/load- c=1&load[]=swfobject.jquery.utils&ver=3.5)
- <http://www.xleesol.com.ar/clientele/space-far.cs.3dsecureclient# password.asp>

- http://www.yahlee.com/phqqwm.phqqwm.phqqwm'*/.hirata.com.mx/img/mps/logo/506.png?alceo.balboni@mail.it
- <http://www.zcyaaale.com/home.html>

The Classification of the Each URL and the Graph is given below

Table 1: Classification of the URL. No 1

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
1	<5	<5	Yes	No	Yes	Malicious

Graph



Figure 2: Classification of URL. No 2

Table 2: Classification of the URL. No 2

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
2	≥ 5	≥ 5	No	Yes	No	Malicious

Graph

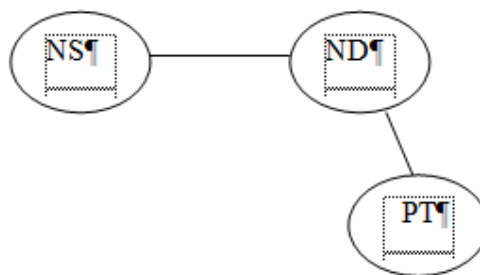


Figure 3: Classification of URL. No 3

Table 3: Classification of the URL. No 3

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
3	<5	<5	No	No	No	Genuine

Graph

Null

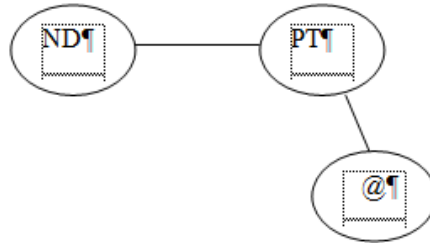


Figure 4: Classification of URL. No 4

Figure 4: Classification of URL. No 3

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
4	<5	≥ 5	No	Yes	Yes	Malicious

Graph

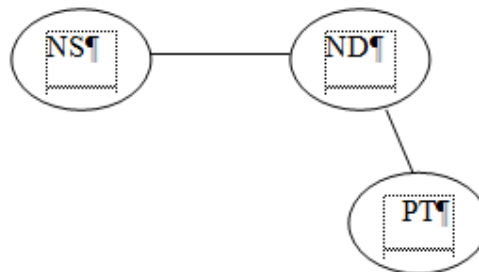


Figure 5: Classification of URL. No 5

Table 5: Classification of the URL. No 5

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
5	≥ 5	≥ 5	No	Yes	No	Malicious

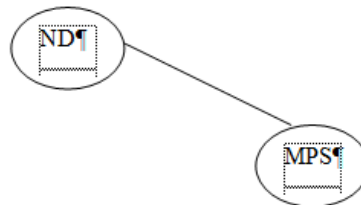


Figure 6: Classification of URL. No 6

Table 6: Classification of the URL. No 6

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
6	<5	≥ 5	Yes	No	No	Malicious

GRAPH:

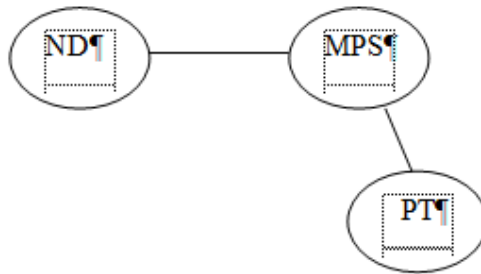


Figure 7: Classification of URL. No 7

Table 7: Classification of the URL. No 7

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
7	<5	≥ 5	Yes	Yes	No	Malicious

Graph

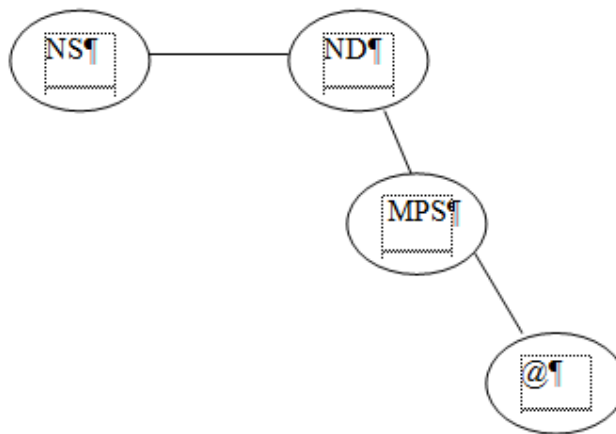


Figure 8: Classification of URL. No 8

Table 8: Classification of the URL. No 8

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
8	≥ 5	≥ 5	Yes	No	Yes	Malicious

Graph

Null

Table 9: Classification of the URL. No 9

URL.NO	No Slashes (NS)	No of Dots in the URL (ND)	Malicious Special Characters (MPS)	Path token (PT)	@ Symbol (@)	Classification
9	<5	<5	No	No	No	Genuine

Graph:

CONCLUSIONS

In this paper, we propose a novel approach to detect malicious URL. The Graph based Classifier, to Detect Malicious URL (GCDMU) is used to identify the malicious URLs. The nodes of the graph represent the malicious features of the URLs. If no malicious feature is present, then the graph is a null graph. It represents that the URL is genuine. Malicious Feature Collect (MFC) algorithm is developed, to collect the list of malicious features from the graph, through traversal. The significant URL features are used for this purpose. The proposed approach is a light weight method and accurately detects malicious URL. Development of a real time behaviour based detection method is our future plan.

REFERENCES

1. Alexandre Gerber, Christopher Dunn, Minaxi Gupta, Oliver Spatscheck, "Prioritizing Malicious website detection", AT&T Intellectual Property I, L.P, Jul 9, 2013.
2. Anjali B. Sayamber and Arati M. Dixit, "Malicious URL Detection and Identification", International Journal of Computer Applications, Volume 99 – No.17, August 2014.
3. APWG, APWG phishing attack trends reports, <http://www.antiphishing.org/resources/apwg-reports>, 2016 [accessed 15.09.16].
4. Ardi, Calvin, and John Heidemann. "AuntieTuna: Personalized Content-based Phishing Detection", USEC '16 Workshop, 21 February 2016, San Diego, CA, USA, 2016.
5. Astorino, A. Chiarello, M. Gaudioso and A. Piccolo, "Malicious URL detection via spherical classification", Journal of Neural Computing and Applications, Volumes 27, 07 June 2016.
6. Charmi Patel, Prof. Hiteishi Diwanji, "A Research on Web Content Extraction and Noise Reduction through Text Density Using Malicious URL Pattern Detection", International Journal of Scientific Research in Science, Engineering and Technology Volume 2, Issue 3, 14 May 2016.
7. Chong, Christophe, et al., "Malicious URL Detection", <http://www.cs.berkeley.edu/~jtma/papers/beyondbl-kdd2009.pdf>, as accessed March 01, 2017.
8. Cuiwen Xiong, Pengxiao Li, Peng Zhang, Qingyun Liu and Jianlong Tan, "MIRD: Trigram-Based Malicious URL Detection Implanted with Random Domain Name Recognition", Proceeding of 6th International Conference on Applications and Techniques in Information Security, November 4-6, 2015.
9. Da Huang, Kai Xu, Jian Pei, "Malicious URL detection by dynamically mining patterns without pre-defined elements", World Wide Web, Volume 17, Issue 6, pp 1375–1394. November 2014.

10. Dohoon Kim, Donghee Choi, and Jonghyun Jin, "Method for Detecting Core Malware Sites Related to Biomedical Information Systems", *Computational and Mathematical Methods in Medicine*, Volume 2015, 17 February 2015.
11. Fuqiang Yu, "Malicious URL Detection Algorithm based on BM Pattern Matching", *International Journal of Security and Its Applications*, Vol.9, No.9 (2015), pp.33- 44.
12. Jyoti Raheja & Sandeep Singh Kang, A Hybrid Scheme for Defending against Malicious Attack in Wireless Sensor Networks, *International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC)*, Volume 3, Issue 4, September - October 2013, pp. 157-162
13. Hiba Zuhair, Ali Selamat and Mazleena Salleh, "New Hybrid Features for Phish Website Prediction", *International Journal of Advances in Soft Computing and its Applications*, Vol. 8, No.1, March 2016.
14. N.Jayakanthan, A.V.Ramani, "A Feature based Framework to Detect Malicious URLs", *International Journal of Control Theory and Applications*, Vol.9, No.9, 2016, pp.1320-1332.
15. N.Jayakanthan, A.V Ramani, "Two phased Classification Model to Detect Malicious URLs", *International Journal of Applied Engineering Research*, Vol.12, No.9, 2017, pp.1893-1898.
16. Jaydeep Solanki, Rupesh G. Vaishnav, "Website Phishing Detection using Heuristic Based Approach", *International Research Journal of Engineering and Technology*, Volume: 03 Issue: 05, May- 2016.
17. Jian Cao, Qiang Li, Yuede Ji, Yukun He, Dong Guo, "Detection of Forwarding-Based Malicious URLs in Online Social Networks", *International Journal of Parallel Programming*, Volume 44, Issue 1 pp 163–180.
18. Prabhu Seshagiri, Anu Vazhayilb, Padmamala Sriram," AMA: Static Code Analysis of Web Page for the Detection of Malicious Scripts ", *Proceedings of the 6th International Conference on Advances in Computing and Communication, ICACC 2016*, 6-8 September 2016, Cochin, India.
19. Ralph Edem Agbefu, Yoshiaki Hori and Kouichi, Sakurai "Domain for the Detection of Malicious Webpages", *International Journal of Cyber-Security and Digital Forensics*, (IJCSDF) 2(2): 36-47 The Society of Digital Information and Wireless Communications, (ISSN: 2305-0012).
20. Roshani K. Chaudhari , D. M. Dakhane, "Machine Learning Approach for Detection of Malicious Urls and Spam in Social Network", *International Research Journal of Engineering and Technology (IRJET)* Volume:03 Issue:05,05May2016.
21. Suyeon Yoo and Sehun Kim ,Two-Phase Malicious Web Page Detection Scheme Using Misuse and Anomaly Detection ,*International Journal of Reliable Information and Assurance* Vol.2, No.1, 2014.
22. Tung-Ming Koo, Hung-Chang Chang, Ya-Ting Hsu and Huey-Yeh Lin , "Malicious Website Detection Based on Honey pot Systems", *2nd International Conference on Advances in Computer Science and Engineering*, CA, USA, July 1-2, 2013.
23. Urvashi Prajapati, Neha Sangal and Deepti Patole, "Fraud Website Detection using Data Mining", *International Journal of Computer Applications*, Volume 141 – No.3, May 2016.
24. Zerina Mašetic, Abdulhamit Subasi, Jasmin Azemovic "Malicious Web Sites Detection using C4. 5 Decision Tree", *Southeast Europe Journal of Soft Computing*, Vol 5 Issue 1 2016.

